Serveur de temp NTP

Rémy De Poorter

2022

Table des matières

1 Introduction

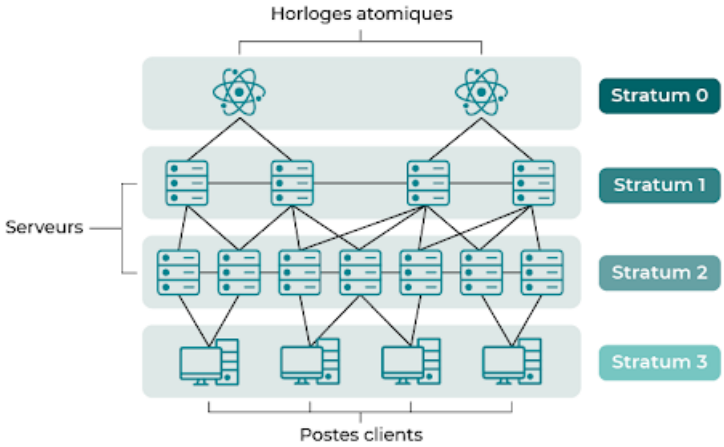
Sur un serveur beaucoup de programmes utilisent l'heure système. Par exemple l'on peut enregistrer l'heure des logs, la connexion d'un utilisateur, l'heure d'une commande sur un site internet. Il faut donc avoir une mesure du temps précise pour permettre la communication entre plusieurs machines. Celles-ci vont utiliser une même mesure du temps pour synchroniser leurs actions

L'on peut configurer manuellement l'heure d'une machine mais ce n'est pas assez précis. C'est pourquoi le protocole NTP (Network Time Protocole) qui est l'un des plus vieux protocoles, est apparu en 1985.

2 Architecture

NTP permet de synchroniser l'heure des différents systèmes à travers un réseau IP. Les clients vont synchroniser leur horloge avec le serveur et chaque serveur se synchronise avec lui-même ainsi que d'autres serveurs. Ce réseau forme donc des couches appelées strata (stratum au singulier). Le plus haut niveau est 0 ce seront des matériels spécialisé couplé avec des horloges atomique. Chaque niveau possède un temps fiable et permet de répartir la demande de temps entre les différents serveurs et d'éviter que l'un soit surchargé.

Une horloge atomique est une horloge qui définit un temps en analysant la fréquence du rayonnement électromagnétique émis par un électron lors du passage d'un niveau d'énergie à un autre. Cela permet d'avoir un temps exact et très stable.



Beaucoup d'organisations gèrent leurs propres serveurs de temps. Certaines n'autorisent qu'une utilisation interne tandis que d'autres autorisent une utilisation publique. Un des plus grands clusters de serveur NTP publics est appelé pool.ntp.org. C'est celui-la qui est configuré par défaut dans la plupart des distributions linux.

3 Faut-il obligatoirement configurer son propre serveur ?

Il est possible d'utiliser le serveur NTP mondial qui est présent par défaut pour synchroniser l'horloge des serveurs mais lorsque le réseau grandit il est intéressant d'avoir son propre serveur NTP car

— La synchronisation entre les serveurs du réseau de l'entreprise sera meilleure avec seulement quelques millisecondes les un des autres..
— Le trafic dû aux synchronisations vers internet sera réduit. Car la définition de l'heure se fera localement et il ne faudra donc pas sortir du réseau local.
— Si un problème technique comme une coupure de courant survient les serveurs restent synchronisés entre eux.
— Cela permet aussi de ne pas dépendre du réseau mondial NTP.

Attention, NTP ne s'occupe pas de tout, en effet l'heure de référence fournie est UTC (temps universel coordonné). Le système d'exploitation aura la tâche de gérer le changement d'heure et les fuseaux horaires en fonction de la position de la machine cliente. Les messages NTP ne sont pas cryptés par NTP et peuvent donc circuler librement sur le réseau. Il est possible de les crypter mais cela va alourdir les paquets à transférer et n'a pas forcément d'intérêt à être sécurisé.

Un serveur NTP est sensible aux attaques de DDOS. Par exemple, un hackeur peut utiliser de fausses adresses IP d'expéditeur et envoyer des paquets au serveur. L'adresse du système embarqué est choisie comme adresse d'expéditeur. Le serveur renvoie sa réponse à l'expéditeur supposé, qui sera la victime. En faisant cela à grande échelle, le système ciblé pourra être surchargé.

4 Les meilleures pratiques

Utiliser des ntp publics pour les hôtes externes :
Si l'entreprise utilise des services, développe des capacités ou d'autres plates-formes intégrées destinées à être déployées hors de l'entreprise, il est envisageable de demander un serveur NTP public à partir du pool de serveur disponible. Il faudrait alors configurer une hiérarchie pour définir les serveurs utilisés pour se synchroniser pour éviter la concurrence d'accès entre le serveur ntp public et celui de l'entreprise.

Configurer son propre serveur ntp interne :
Il est possible d'acheter des appliances NTP Stratum 1 ou 0 à utiliser en interne, ou de mettre en place son propre serveur NTP pour un faible coût.

Standardiser sur le temps UTC :
Normaliser tous les appareils au système universel de temps UTC permet de simplifier la corrélation des journaux entre l'organisation et les parties externes, quel que soit le fuseau horaire utilisé par l'appareil.

Sécuriser le service de réseau de temps :
Il est recommandé de restreindre les commandes utilisables sur le serveur stratum, de ne pas autoriser les requêtes publiques des serveurs de strate et donc d'autoriser uniquement les réseaux connus.

Cryptage :
Il existe des services de cryptage associés à NTP mais le cryptage n'est pas toujours nécessaire. L'entreprise en a-t- elle vraiment besoin et à quel niveau de complexité ? Plus un échange est crypté, plus il sera sécurisé mais consomme aussi davantage de ressources et apporte avec lui des sources de problèmes potentiels supplémentaires.

Lois de Segal :
Avoir plusieurs sources de temps permet de maintenir une heure précise même en cas de panne de l'un des serveurs. La loi de ségal dit que si on a deux serveur d'horodatage avec une heure différente on ne peut pas savoir la quelle est la plus précise. C'est pourquoi il est conseillé d'avoir au moins 3 serveurs stratum 0 ou stratum 1 et de les utiliser comme maîtres principaux. Il est aussi conseillé de répartir géographiquement les différents serveurs.

Gérer le nombre de clients :
En cas de forte utilisation des serveurs de temps, l'on peut ajouter des stratum secondaire pour augmenter la bande passante et réduire la charge de travail des serveurs principaux.

Rester à jour.
Il est conseillé de faire les mises à jour du serveur de temps pour combler les failles. Et donc être mieux protéger des attaques réseau.

5 Mise en place

5.1 Procédure pour modifier manuellement l'heure du sytème

Afficher l'heure avec systemd :

```
sudo timedatectl
```

```
user0@localhost:~> sudo timedatectl
[sudo] Mot de passe de root :
      Local time: jeu 2022-03-17 10:47:17 CET
  Universal time: jeu 2022-03-17 09:47:17 UTC
        RTC time: jeu 2022-03-17 10:47:18
       Time zone: Europe/Brussels (CET, +0100)
 Network time on: yes
NTP synchronized: yes
 RTC in local TZ: yes
```

Définir l'heure au format année mois jour heure minute seconde :

```
sudo timedatectl set-time "2015-11-20 16:14:50"
```

Afficher les fuseau horaires disponibles avec un filtre :

```
sudo timedatectl list-timezones | grep Brussels
```

Définir le fuseau horaire :

```
sudo timedatectl set-timezone | Europe/Brussels
```

Voir la synchronisation de l'heure :

```
sudo systemctl status systemd-timesyncd.service
```

```
user0@localhost:~> sudo systemctl status systemd-timesyncd.service
[sudo] Mot de passe de root :
● systemd-timesyncd.service - Network Time Synchronization
   Loaded: loaded (/usr/lib/systemd/system/systemd-timesyncd.service; enabled; >
   Active: active (running) since Thu 2022-03-17 10:46:45 CET; 15min ago
     Docs: man:systemd-timesyncd.service(8)
 Main PID: 597 (systemd-timesyn)
   Status: "Idle."
    Tasks: 2
   CGroup: /system.slice/systemd-timesyncd.service
           └─597 /usr/lib/systemd/systemd-timesyncd
```

5.2 Procédure pour mettre en place un serveur utilisant le pool de serveur ntp

Configurer le serveur

connaitre l'adresse ip du serveur :

```
sudo ip a
```

Sur OpenSuse le serveur NTP par défaut s'appelle chrony, l'installer si ce n'est pas déjà fait à l'aide de :

```
sudo zypper install chrony
```

Pour configurer chrony, il faut éditer le fichier de configuration de celui-ci :

```
sudo nano /etc/chrony.conf
```

```
# Use public servers from the pool.ntp.org project.
# Please consider joining the pool (http://www.pool.ntp.org/join.html).
pool 0.opensuse.pool.ntp.org iburst
pool 1.opensuse.pool.ntp.org iburst
pool 2.opensuse.pool.ntp.org iburst
pool 3.opensuse.pool.ntp.org iburst
! pool pool.ntp.org iburst

# Record the rate at which the system clock gains/losses time.
driftfile /var/lib/chrony/drift

# Allow the system clock to be stepped in the first three updates
# if its offset is larger than 1 second.
makestep 1.0 3

# Enable kernel synchronization of the real-time clock (RTC).
rtcsync

# Enable hardware timestamping on all interfaces that support it.
#hwtimestamp *

# Increase the minimum number of selectable sources required to adjust
# the system clock.
#minsources 2

# Allow NTP client access from local network.
#allow 192.168.0.0/16

# Serve time even if not synchronized to a time source.
#local stratum 10

# Specify file containing keys for NTP authentication.
#keyfile /etc/chrony.keys

# Get TAI-UTC offset and leap seconds from the system tz database.
#leapsectz right/UTC

# Specify directory for log files.
logdir /var/log/chrony

# Select which information is logged.
#log measurements statistics tracking

# Also include any directives found in configuration files in /etc/chrony.d
include /etc/chrony.d/*.conf
```

pool définit les serveurs utilisés pour se synchroniser par défaut c'est le site de opensuse.

driftfile est le fichier dans lequel chrony va noter la différence de temps entre l'horloge locale et celle des serveurs de référence.

rtcsync informera le noyau que l'horloge système est synchronisée et que le noyau mettra à jour l'horloge RTC toutes les 11 minutes. Donc commenter rtcsync

Par défaut chrony n'autorise pas les clients à se synchroniser avec notre serveur, il faut donc les autoriser en ajoutant la ligne suivante contenant le réseau du serveur avec allow 192.168.0.0/16

Ajouter un stratum local qui sera utilisé en cas de non connexion au serveur publics.
local stratum 10

ensuite quitter en enregistrant les modifications.
Configurer le démarrage automatique de chrony avec le système :

```
sudo systemctl enable chronyd
```

Démarrer le serveur :

```
sudo systemctl start chronyd
```

Pour gérer le serveur chrony on utilise l'interface en ligne de commande chronyc afficher la liste des serveur ntp auquel le nôtre est synchronisé :

```
sudo cronyc sources
```

* pour le serveur utilisé comme référence
+ pour les serveur qui servent à calculer une moyenne de temps
- pour les serveurs actuellement non utilisés.

```
user0@localhost:~> sudo chronyc sources
210 Number of sources = 4
MS Name/IP address         Stratum Poll Reach LastRx Last sample
===============================================================================
^+ ntp.devrandom.be              2   9   377   130    -12us[-8509ns] +/-   10ms
^* ntp2.belbone.be               2   9   377   129   -148us[ -144us] +/- 9268us
^- dns-rec-2-brudie.belnet.>     2   7   377   127   -739us[ -739us] +/-   42ms
^+ time.cloudflare.com           3   7   377    69  +1528us[+1528us] +/-   15ms
user0@localhost:~>
```

Configurer le client éditer le fichier de configuration :

```
sudo nano /etc/systemd/timesyncd.conf
```

ajouter l'ip du serveur

```
  GNU nano 4.9.2              /etc/systemd/timesyncd.conf
#  This file is part of systemd.
#
#  systemd is free software; you can redistribute it and/or modify it
#  under the terms of the GNU Lesser General Public License as published by
#  the Free Software Foundation; either version 2.1 of the License, or
#  (at your option) any later version.
#
# Entries in this file show the compile time defaults.
# You can change settings by editing this file.
# Defaults can be restored by simply deleting this file.
#
# See timesyncd.conf(5) for details.

[Time]
NTP=192.168.126.133/24
#FallbackNTP=0.opensuse.pool.ntp.org 1.opensuse.pool.ntp.org 2.opensuse.pool.nt
```

redémarrer systemd :

```
sudo systemctl restart systemd-timesyncd
```

redémarrer le chronyd :

```
sudo systemctl restart chronyd
```

Afficher les sources utilisées par chrony :

```
chronyc sources
```

L'étoile correspond au serveur utilisé

```
user0@install:~> chronyc sources
210 Number of sources = 5
MS Name/IP address         Stratum Poll Reach LastRx Last sample
===============================================================================
^* 10.0.255.14                   4   6    17    34   +790ns[ +134ns] +/-   14ms
^- ntp.vives.be                  3   6    17    39  -6425us[-6425us] +/-   15ms
^- ntp.devrandom.be              2   6    17    39  -6933us[-6933us] +/-   11ms
^- ntp2.unix-solutions.be        2   6    17    39  -8622us[-8623us] +/-   30ms
^- webserver.discosmash.com      2   6    17    38  +2171us[+2171us] +/-   66ms
user0@install:~> chronyc tracking
Reference ID    : 0A00FF0E (10.0.255.14)
Stratum         : 5
Ref time (UTC)  : Fri Mar 11 13:14:38 2022
System time     : 0.000001015 seconds slow of NTP time
Last offset     : +0.001221616 seconds
RMS offset      : 0.001221616 seconds
Frequency       : 15.750 ppm fast
Residual freq   : +16.514 ppm
Skew            : 9.106 ppm
Root delay      : 0.023527831 seconds
Root dispersion : 0.004874270 seconds
Update interval : 64.7 seconds
Leap status     : Normal
user0@install:~>
```

5.3 Procédure pour créer un serveur ntp sans utiliser de pool public

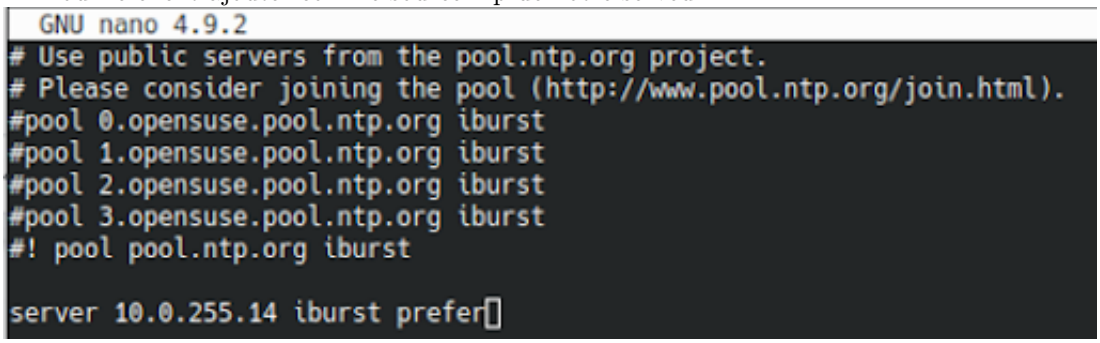
Pour le serveur suivre la procédure précédente mais changer ceci dans le fichier de configuration de chrony

```
sudo nano /etc/crony.conf
```

commenter les pools originaux car on en a plus besoin

définir un stratum local local stratum 10 allow 10.0.255.0/24

Pour le client ajouter comme source l'ip de notre serveur



Redémarrer chronyd

```
sudo systemctl restart chronyd
```

6 Procédure pour planifier une tâche avec cron.

autoriser l'utilisateur à exécuter la tâche :

```
sudo nano /etc/cron.allow
et y ajouter l'utilisateur (user0)
```

ajouter une tâche :

```
sudo crontab -e
```

ajouter les paramètres de temps ainsi que la commande à exécuter minute (1 à 60) heure (1 à 24) jours (1 à 31) mois (1 à 12 ou leur libellé anglais jan, feb, mar) jour (1 à 7 ou leur libellé anglais mon, tue, wed) commande à lancer une * est utilisée si on ne renseigne rien */1 veut dire que on lance la commande toutes les minutes si on avait seulement mis 1 alors la commande s'exécutera à chaque première minute de chaque heure (00h01, 01h01, 02h01, ...)

*/1 * * * * echo "coucou" » remy.txt

7 Alternative a NTP le SNTP

Une version simplifiée du protocole NTP est développée en parallèle. Elle s'appelle SNTP pour Simple Network Time Protocole. Elle est destinée à des réseaux ou la précision à la seconde suffit. C'est une version qui allège avec des algorithmes avec un traitement de paquets plus léger. Il est utilisé pour les systèmes embarqués où la capacité de calculs est très limitée. Cette implémentation du temps plus simple dialogue quand même avec des serveurs NTP standards. SNTP doit donc être utilisé que quand c'est nécessaire pour ne pas perturbé et fausser le réseau NTP. Il est également recommandé de n'utiliser SNTP que sur des systèmes clients.

8 Planificateur de tâche CRON

Cron est un programme qui va exécuter automatiquement des tâches à une date et une heure spécifiées (le vendredi 13 janvier 2022) ou selon un cycle (tous les ans). Cron est aussi appelé le planificateur de tâche ou gestionnaire de tâche planifiées. Les tâches sont définies dans le fichier /etc/crontab. Elles s'exécutent avec les droits de root sans demander le mot de passe. Pour son utilisation voir procédures.

9 Conclusion

Le protocole NTP est très utilisé à travers le monde pour la synchronisation des machines. Son succès est lié à sa facilité de mise en place avec plus de 20 ans d'expérience et de recherches il est donc très fiable. Les réseaux NTP ont révolutionné les méthodes de traitement des paiements, la sécurité en ligne en évitant que le client soit réglé à une heure différente de celle du serveur. On pourrait imaginer une prochaine version du protocole permettant encore plus de fiabilité et de précision pour déterminer le temps même en attendant la version actuellement utilisée en fait déjà un protocole incontournable.

10 Bibliographie

Références

[1] https ://doc.opensuse.org/documentation/leap/archive/15.0/reference/html/book.opensuse.reference/cha.ntp.html

[2] https ://chrony.tuxfamily.org/index.html

[3] https ://doc.ubuntu-fr.org/ntp

[4] https ://ubuntu.com/server/docs/network-ntp

[5] https ://www.pool.ntp.org/zone/be

[6] https ://chrony.tuxfamily.org/examples.html

[7] https ://help.uis.cam.ac.uk/service/network-services/network-time-protocol-ntp/details-of-the-network-time-protocol-service

[8] https ://insights.sei.cmu.edu/blog/best-practices-for-ntp-services/

[9] https ://services.renater.fr/ntp/article/presentation_ntp_article

[10] https ://www.cisco.com/c/fr_ca/support/docs/availability/high-availability/19643-ntpm.html

[11] http ://www.audentia-gestion.fr/cisco/pdf/1094258_ntpm.pdf

[12] http ://www.audentia-gestion.fr/cisco/pdf/1094258_ntpm.pdf

[13] https ://docs.microsoft.com/fr-fr/windows-server/networking/windows-time-service/how-the-windows-time-service-works

[14] https ://en.opensuse.org/SDB :Cron

11 Annexes

Les commandes

Affiche les informations sur l'heure de votre système :

```
timedatectl
```

Installer le logiciel chrony comme serveur ntp :

```
sudo zypper install chrony
```

Afficher l'adresse ip de la machine :

```
ip a
```

Éditer le fichier de configuration de chrony :

```
sudo nano /etc/chrony.conf
```

Lancer chrony au démarrage :

```
sudo systemctl enable chronyd
```

Démarrer ou redémarrer chrony :

```
sudo systemctl start chronyd
```

Vérifier le statut de chronyd :

```
systemctl status chronyd
```

Désactiver la synchronisation ntp gérer par systemd :

```
sudo timedatectl set-ntp false
```

Vérifier les sources utilisées par chrony :

```
sudo chronyc sources
```

Éditer le fichier de configuration de timesyncd :

```
sudo nano /etc/systemd/timesyncd.conf
```

(Re)Démarrer le service systemd :

```
sudo systemctl restart systemd-timesyncd
```

Vérifier le statut de systemde :

```
sudo systemctl status systemd-timesyncd
```

Désactiver le lancement automatique de chrony au démarrage du système :

```
sudo systemctl disable chrony
```

Stop chrony :

```
sudo systemctl stop chrony
```

Définir l'heure au format année mois jour heure minute seconde :

```
sudo timedatectl set-time "2015-11-20 16:14:50"
```

Pour afficher les fuseau horaires disponibles :

```
sudo timedatectl list-timezones
```

Pour afficher les fuseau horaires disponibles à l'aide d'un filtre :

```
sudo timedatectl list-timezones | grep Kiev
```

Autoriser un user à exécuter des tâches programmées :

```
sudo nano /etc/cron.allow
et y ajouter l'utilisateur (user0)
```

Interdire à un user à exécuter des tâches programmées :

```
sudo nano /etc/cron.deny
et y ajouter l'utilisateur (user0)
```

Pour définir une tâche nous allons éditer le fichier :

```
sudo crontab -e
```

ensuite l'on défini le moment ou la commande va être exécutée : minute (1 à 60) heure (1 à 24) jours (1 à 31) mois (1 à 12 ou leur libellé anglais jan, feb, mar) jour (1 à 7 ou leur libellé anglais mon, tue, wed) commande à lancer une * est utilisée si on ne renseigne rien */1 veut dire que on lance la commande toutes les minutes si on avait seulement mis 1 alors la commande s'exécutera à chaque première minute de chaque heure (00h01, 01h01, 02h01, ...)

Ce qui donne par exemple pour écrire coucou toutes les minutes dans le fichier remy.txt */1 * * * * echo "coucou" » remy.txt

Pour voir qu'elle s'est bien enregistrée on peut afficher la liste des tâche :

```
sudo crontab -l
```

12 Vocabulaire

CRON
Est un planificateur de tâche

Chrony
Est une implémentation versatile (qui change facilement) du protocole NTP.

Chronyd
Est un démon exécuter dans l'espace utilisateur

Chronyc
Est un outil en ligne de commande pour gérer chronyd

BURST et IBURST
Ils permettent de calibrer initialement et rapidement une horloge système.
Burst envoie des paquets toutes les 2 secondes et IBURST toutes les 16 secondes pour synchroniser les horloges.

BURST envoie une rafale de 8 paquets lorsque le serveur est accessible et est utilisé afin de mesurer avec précision la gigue.

IBURST envoie lui aussi une rafale de 8 paquets mais dans ce cas si le serveur est inaccessible et raccourci les délais jusqu'à la première synchronisation. Spécifier IBURST permet donc d'avoir une synchronisation d'horloge plus rapide même si ce mode est considéré comme agressif envers certains serveur NTP publics car si le serveur ne répond pas le mode IBURST continue d'envoyer des requêtes fréquentes jusqu'à ce que le serveur réponde et que la synchronisation de l'heure démarre.

La gigue
Ce nombre est une valeur absolue en millisecondes, indiquant l'écart quadratique moyen de vos décalages.

Driftfile
Il est utilisé pour stocker le décalage de fréquence entre l'horloge système locale et celle de fréquence requise pour rester synchronisée avec l'heure UTC. Le fichier de dérive est lu pendant le démarrage système et sa valeur est utilisée pour corriger la source horaire. Utiliser ce fichier de dérive réduit le temps nécessaire pour atteindre une heure stable et précise. La valeur est calculée et le fichier est remplacé une fois par heure par ntpd.Le fichier de dérive est remplacé plutôt que simplement mis à jour c'est pourquoi ntpd possède l'autorisation d'écriture sur ce fichier.

RTC (Real Time Clock)
Horloge de temps réel, l'horloge système.

RTCSYNC

rtcsync informera le noyau que l'horloge système est synchronisée et donc le noyau mettra à jour l'horloge RTC toutes les 11 minutes. Si aucun rtcfile ou rtcsync n'est dans la configuration, chronyd ne se souciera pas du tout du RTC. Il ne suivra pas son décalage de fréquence par rapport à l' heure système et ne corrigera pas la dérive avec l'option -s, ou ne demandera pas au noyau de le synchroniser. Si le RTC dérive, l'heure initiale de démarrage sera erronée.